(19) Europäisches Patentamt
European Patent Office
Office européen des brevets

(11) EP 1 146 472 A2

(12) EUROPEAN PATENT APPLICATION

(43) Date of publication:
17.10.2001 Bulletin 2001/42

(51) Int Cl.[7]: **G06K 9/64**

(21) Application number: **01103877.5**

(22) Date of filing: **16.02.2001**

(84) Designated Contracting States:
AT BE CH CY DE DK ES FI FR GB GR IE IT LI LU MC NL PT SE TR
Designated Extension States:
AL LT LV MK RO SI

(30) Priority: **17.02.2000 US 505875**

(71) Applicant: **Xerox Corporation, Patent Department**
Rochester, New York 14644 (US)

(72) Inventor: **Handley, John C.**
Fairport, NY 14450 (US)

(74) Representative: **Grünecker, Kinkeldey,**
Stockmair & Schwanhäusser Anwaltssozietät
Maximilianstrasse 58
80538 München (DE)

(54) Loose-gray-scale template matching

(57) An apparatus and method of template fitting for fitting a plurality of templates in correspondence with a gray-scale image, including receiving gray-scale image information of an observed image; defining a window of the gray-scale image information so as to enable a loose-grayscale fit between the defined window and the plurality of templates; extracting a plurality of pixels of the image information and using the window to which to fit at least one of the plurality of templates. Further, fitting one of the plurality of templates to the defined window within a threshold looseness interval associated with the template wherein a looseness interval between the window of image data and the template is less than the threshold looseness interval

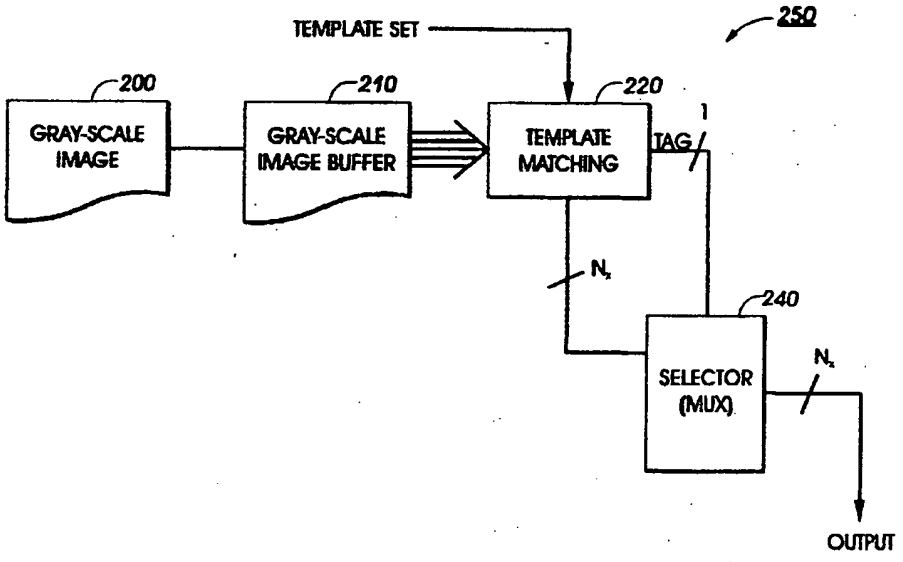

FIG. 6

EP 1 146 472 A2

Description

BACKGROUND OF THE INVENTION

Field of Invention

[0001] This invention relates to systems and methods for processing images using filters. More specifically, this invention relates to systems and methods for designing and implementing image processing filters using templates wherein the filters operate on gray-scale images and the templates identify gray-scale image features for the purposes of modification or for extracting some image statistic, and for the purposes of optimization for the human visual system, or compatibility with other system modules, such as, compression algorithms, recognition algorithms and those occurring in printing and display devices.

DESCRIPTION OF RELATED ART

[0002] A wide variety of digital document processing tasks are performed using template-based filters. Illustratively, digital document processing tasks include resolution conversion, enhancement, restoration, appearance tuning and de-screening of images. These tasks are commonly performed on monochrome and color images, as well as binary and continuous tone images. Although, due to the binary nature of conventional templates, implementing many digital document-processing tasks on continuous tone images has been problematic prior to the present invention. A continuous tone image may also be referred to as a gray-scale image.

[0003] In conventional systems and methods, a typical filter includes template operators to perform filtering of the images, where, a filter may be characterized as an operator or device that transforms one image into another image or transforms an image to a collection of information, such as image statistics. The filter is formed of a number of imaging template operators, often simply referred to as templates. These templates may be, for example, stored in a look-up table and implemented using a look-up table formalism. Or other equivalent formalisms, such as Boolean logic may be employed. The number of templates in a filter may vary between a small number of templates to thousands of templates. Due to its versatility in design, a look-up table is typically used to implement a template-based filter.

[0004] A raster is a one-dimensional array of image data, reflecting a single line of data across a single dimension, i.e., the length or the width, of the image. Further, each location, or "picture element," in an image may be called a "pixel." In an array defining an image in which each item of data provides a value, each value indicating the properties of a location may be called a pixel value. Each pixel value is a bit in a binary form of an image, a gray-scale value in a gray-scale form of an image, or a set of color-spaced coordinates in a color coordinate form of an image. The binary form, gray-scale form, and color coordinate form are each arranged typically in a two-dimensional array, which defines an image. An N-dimensional array is typically used for an N-dimensional images, where for example, $N = 3$ for 3-dimensional topographic images.

[0005] Using the typical binary image processing setting as an example, the filter, using the templates, transforms certain observed pixel patterns in a binary image, for example, into a corresponding enhanced binary pixel pattern. Specifically, the filter observes an arrangement of pixels using a suitable window or mask. A window is an imaging algorithmic device that observes a plurality of pixels at the same time, where the plurality of pixels is located about a target pixel. The values and locations of the observed pixels are inputted into the template matching operations. After observing the arrangement of pixels, about a target pixel, the filter then attempts to match the observed pixel pattern with one or more of the templates in the look-up table. If the look-up table contains a match to the observed pixel pattern, the look-up table generates an appropriate output. The output may be an enhanced pixel pattern for the target pixel that corresponds to the observed pixel pattern. The output could also be information in other forms; for example, the output could be a code denoting the match condition, or a data to be used for a statistical characterization of image regions.

[0006] A wide variety of types and sizes of observation windows or masks are known. The particular window used in a particular application depends on the image to be analyzed and the particular process to be performed on the image. Illustratively, a 3x3 window may be used to process an image. The 3x3 window, at various locations in the image, observes a 3x3 block, i.e., a 9-pixel block, of binary-valued pixels, for example. One pixel in the window is the target pixel, which is typically the center pixel, while the other pixels in the window are the neighboring pixels. The target pixel and the neighboring pixels form a neighborhood. The window is typically scanned across an image advancing from target pixel to target pixel.

[0007] After the neighborhood is observed in the window, the neighborhood is then processed in some manner. For example, the observed neighborhood may be transformed into a vector. The vector is expressed in the form of $(x_1, x_2...x_N)$ where N is the number of pixels in the neighborhood and is used to represent the properties of the target pixel, including the neighborhood of the target pixel. Each element of the vector represents one of the pixels observed in the

window. The vector is then used in the look-up table to generate a desired output, for example.

[0008] A look-up table may be created in a wide variety of ways. Typically, an input value is input into the look-up table and, in response, the look-up table outputs an output value. Further, the look-up table is typically created using a training image or a set of training images. "Restoration and Enhancement of Digital Documents," by R. Loce and E. Dougherty, teaches a variety of methods for designing templates based on sets of training images. The training images will occur in pairs, where one member is the "typically input image," or the "typically observed image," i.e., the "observed image," and the other image is the "ideal desired processed version of the image," i.e., the "ideal image." The training image pairs may be input into a computer program that acquires and analyzes pattern statistics between the two images, i.e., using computer-aided filter design techniques.

[0009] Conventional computer-aided filter design may be accomplished through using training-sets of document bitmaps, for example.

[0010] Illustratively, for designing a filter that enhances from a binary state to a gray-scale state, for a given pattern that occurs in the binary image about a target pixel, a training analysis system examines a target pixel at that corresponding location in the gray-scale image. The center of the window may be placed at the target pixel, for example. Based on the set of gray-scale pixels in the gray-scale image that are associated with corresponding target pixels in the binary image and gray-scale image, and associated with a similar neighborhood pixel pattern, a "best gray-scale pixel value" is determined for processing a target pixel that possess that pattern of neighborhood pixels. In other words, a template is created for the target pixels in the binary image possessing similar neighborhood pixel patterns. This analysis is performed for all binary patterns that are significant.

[0011] In this process of template selection, significance may be due to attributes such as the pixel pattern's frequency of occurrence, the pattern's effect on the generated image, or both. Accordingly, if a template, i.e., a pattern of pixels in the binary image, is considered significant with respect to template inclusion in the design process, that template will appear in the template-matching filter. Upon operating on an input image, if that pattern is observed, the observed target pixel value will be assigned or associated with a certain value, i.e., a corresponding gray-scale value. Both the observed neighborhood and the corresponding gray-scale value may be stored in the look-up table. Accordingly, the look-up table accepts input values and outputs a desired corresponding output value, i.e., maps input values to an ideal corresponding output value.

[0012] However, it should be apparent that this input/output process may be performed in various other ways without using a look-up table. One alternative approach that is equivalent to using a look-up table representation is a Boolean logic representation. In the Boolean logic representation, pixel values are used as variables in the logic architecture, such as a logical sum of products. The goal of template filter design using Boolean logic representation is to derive optimized Boolean operators, preferably statistically optimized Boolean operators.

[0013] As illustrated in Fig. 1 there is shown the basic process for template matching based on an observed image. Initially an observed image will occur (10) from which an ideal image is warranted (15). In order to generate an image as close as possible to the ideal image, a generated image (25) is created by utilizing a template matching operation (20). With the above understanding of the image processing setting we see that it is greatly desired to produce a generated image as similar as possible to the ideal image.

[0014] The conventional template matching operation is a binary matching process whereby a binary template or filter is to applied to a binary image.

[0015] For example, as illustrated in Fig. 2A a binary image (50) is initially sent to a data buffering circuit (55). After which, a binary pattern matching operation (60) is performed. Binary templates are typically defined as possessing ones, zeros, and "don't cares." With reference to Fig. 2B there is shown a typical binary matching template structure. For example, the binary templates in this example are defined with ones and X's for use in the binary matching step, where "X" denotes "don't care". After combining the buffered image data and the templates in the binary pattern matching operation (60), an enhanced data buffering step (65) to create the enhanced image (70) is exercised.

[0016] Illustrated in Figs. 4A, 4B, and 4C is the set of images associated with an image restoration application using a filter defined by templates representing particular Boolean functions applied to a received image. Fig. 3A shows a representation of a 3x3 window and the positions of pixels $(x_1,...,x_9)$. The filter defined by this window and the templates of Fig. 2B possesses the Boolean function representation shown in Fig. 3B as follows:

$$(1) \qquad y=f(x_1, x_2, ..., x_9) = x_5 + x_4\, x_6 + x_1\, x_9 + x_2x_8+x_3x_7$$

[0017] When employed as a translation-invariant filter, the singleton template x_5 behaves as an identity operator: whatever pixels are valued one and zero in the input image are valued one and zero in the output image, respectively. ORed onto that identity image is the result of each 2-pixel logical product. The templates corresponding to those products possess structures that straddle the origin pixel. In this configuration a template can "fit", or yield a 1, when positioned about a hole or break in a character stroke. Equation 1 is an example of an image processing operator that can

be employed to repair breaks in character strokes within an image. Fig. 4 is an example of applying Equation 1 as a filter operating on an image, where Fig. 4A shows an ideal image, 4B shows an input image, and 4C shows the image resulting from applying Equation 1 as a filter to the image of Fig. 4B., The filter defined by Equation 1 produced a character (120) more likely to be recognized in character recognition operation. The restoration is not perfect. A more complicated (more and different products) filter could achieve better restoration. As shown in Fig. 4A and Fig. 4C there are still differences between the ideal image 100 and the generated image 120 resulting from the use of the filtering process defined by Equation 1.

[0018] As illustrated in Fig. 5A a system of the prior art is shown where an input image 150 is processed by a process 165 that includes binary-template-based filters 160 and combinatorial logic 170 [changed t to match the figure]; from which an output image 175 is generated and subsequently sent to a digital printer, high resolution printer or high resolution display 180.

SUMMARY OF THE INVENTION

[0019] Loose-gray-scale template matching is a windowed operation (filter) that operates on gray or partial gray images and signals. In some ways it can be thought of as an extension of the widely practiced binary template matching methods into gray-scale. Template matching operations performed on binary data (line width control, resolution enhancement, document clean up, etc.) should also be performed on partial gray imaged obtained via antialiasing or scanning. Exact matching in gray-scale would often require a prohibitively large number of templates, hence loose fitting has been developed so that a range image signals could be matched by a single template. As with binary template matching, loose-gray-scale template matching would typically be performed using groups of templates where each template or subgroup of templates is associated with a particular output or action.

[0020] In addition to extending binary template matching applications to gray-scale, the loose-gray-scale template matching method introduces an additional tool that can well address other significant image process problems such as enabling a pass through mode for scanned halftones, feature recognition, and segmentation. As with other windowed operators, loose-gray-scale template matching can be performed adaptively and in several dimensions (e.g., 2-dimensional spatial images, 3-dimensional topological images, multiple color planes, frames in a time sequence).

[0021] With respect to a preferred statistically optimized design procedure, an observed image and the ideal image may be received in the form of digital images. In accordance with conventional methods for designing a filter and the templates with the filter, the design method tries to map as accurately as possible the observed image to an image that accurately corresponds to the ideal image given certain constraints such as limited number of templates.

[0022] With reference to Fig. 1, the relationship of an observed image, an ideal image, matching templates and a generated image is shown. The observed image and the ideal image represent a typical "before" and "ideal after" set of images, respectively. In other words, the observed image and the ideal image can be utilized as training images for a filter design procedure. The observed image and the ideal image may be actual images that are selected based on expert knowledge, or they may contain particular structures that are useful to training. For instance, they may contain precise patterns of particular concern to a specific image processing task.

[0023] To further explain, in the filtering operation, given an observed image, in some cases it is possible to generate an image that is identical to the ideal image. Typically, the generated image will not be an exact replica of the ideal image because the filter operation would require knowledge that may not be present. In addition, exact generation of the ideal image would most likely involve exact matching of patterns in the gray-scale image. Such matching is impractical due to the large number of templates and processing time that would be required for most applications. This situation is the result of time constraints, memory constraints, and other system and resource constraints. Further it is recognized that the human visual system has certain limitations in what it can perceive. The excessive computation and storage required for exact pattern matching in gray-scale may yield results that are beyond the perceptive capabilities of the human visual system, or in general, beyond the limitations of the system that will receive the image. Given the above practical system constraints and limitations, a filtering approach using loose-gray-scale template matching has been proposed. In a practical setting with realistic limitations the loose-gray-scale template matching process allows a certain number of templates to have a better match with an observed image to form a generated image closer to the ideal image, than would an attempt to utilize the same number of exact matching templates on the same gray-scale image. If practical limitations with realistic limitations do not allow for many templates, the results of exact matching on gray-scale images would generally be very poor. This new process minimizes costs and generating time of the ideal image, and allows the observer to focus on details that the human visual perception and subsequent image processing modules will actually detect. In addition, the burden of designing an impractical number of exact match templates is mitigated by designing a much smaller set of loose fitting templates.

[0024] The present invention separately provides the methods that may be used in conjunction with known methods for line width modification, resolution conversion, resolution enhancement, image restoration and the like.

[0025] The present invention separately provides systems and methods that may be used in conjunction with filter

design and implementation techniques to optimally yield images processed to better visual quality.

[0026] In accordance with one embodiment of the system and methods of the present invention there is provided a template fitting method for fitting a plurality of templates in correspondence with a gray-scale image,, comprising:- receiving gray-scale image information of an observed image; defining a window of said gray-scale image information so as to enable a loose-gray scale fit between said defined window and said plurality of templates; extracting a plurality of pixels of said image information using said window to which to fit at least one of the plurality of templates; and fitting one of the plurality of templates to said defined window within a threshold looseness interval associated with the template wherein a looseness interval between the window of image data and the template is less than the threshold looseness interval.

A further embodiment comprises, generating using processing circuitry a looseness interval between window of received image and each template associated therewith; and comparing the threshold looseness interval to the looseness interval using the processing circuitry.

[0027] In a further embodiment the template fitting method comprises fitting said plurality of templates either in a parallel or serial process.

In a further embodiment the processing circuitry includes algorithmic operations for associating the templates with the gray-scale image information.

In a further embodiment the processing circuitry stores a plurality of pixels of the window of received image information which includes image pixels surrounding a target pixel.

[0028] In accordance with another embodiment of the system and methods of the present invention there is provided a multiple template fitting method including a plurality of single templates or plurality of families of templates for correspondence with a gray-scale image , including receiving image information of an observed image; defining a window of the image information so as to enable a loose grayscale fit between the defined window and the plurality of singular or families of templates; and fitting one of the plurality of singular or families of templates to the defined window within a threshold looseness interval and outputting an item of data indicating a result of a correspondence between a single or family of templates and the received image information within a threshold looseness interval.

[0029] In a further embodiment said templates and said threshold looseness intervals are stored in a representation which enables fitting one of a plurality of singular or families of templates to said defined window.

[0030] In a further embodiment said representation is either multiple nominal pixel values with incremental corresponding upper and lower threshold values, a lower value and upper value associated with the threshold interval, a reduced quantization state with respect to a quantization state of the received image.

[0031] According to a further aspect, a method for matching a plurality of templates with a received image wherein said received image being bitmap data including at least a plurality of gray-scale pixel tiles that define said image, said matching method comprises:

receiving said image data so as to extract pixel tile information of said received image wherein said pixel tile information being of a predetermined matrix size, matching loosely said pixel tile information with at least one of a plurality of templates so as to generate pixel-wise looseness interval values therebetween, and outputting a portion of enhanced said pixel tile information wherein said enhanced pixel tile information being formed by a matching of a template with pixel-wise looseness values.

A further embodiment comprises storing template pixel tile information and said pixel-wise looseness values.

A further embodiment comprises storing said templates and template-wise looseness values.

[0032] In a further embodiment said templates and said pixel-wise threshold looseness interval is stored in a representation which enables fitting one of a plurality of single or families of templates to said pixel tile information.

[0033] In a further embodiment said representation is either multiple nominal pixel values with incremental corresponding upper and lower threshold values, a lower value and upper value associated with the threshold interval, a reduced quantization state with respect to a quantization state of the received image.

In a further embodiment the method comprises using arithmetic or algorithmic processes to calculate the pixel-wise looseness interval values.

In a further embodiment the method of comprises using arithmetic or algorithmic processes to calculate the template-wise looseness values.

In a further embodiment the method comprises combining said pixel-wise looseness values to form template-wise looseness values.

In a further embodiment the outputted enhanced tile information is grayscale or high addressability information.

In a further embodiment said matching step generates a match identifier to match said pixel tile information.

In a further embodiment said matching step generates a match identifier to match said templates.

In a further embodiment the matching step outputs an item of data indicative of a match performed by the matching step.

[0034] In accordance with another embodiment of the system and methods of the present invention there is provided a loose-gray-scale template matching method including at least an initial point for locating within the received image a plurality of pixel points surrounding said initial point, the method including the steps of: locating said initial point and

said plurality of pixel points, generating a looseness interval about said located initial point with template information associated therewith, determining which one of a plurality of templates for fitting fits within a threshold looseness interval and, outputting an item of data indicative of whether a template is within the looseness threshold thereby indicating that the template is in agreement therewith.

A further embodiment comprises combining the pixel-wise looseness intervals to form a template-wise looseness interval.

A further embodiment comprises in said determining step using arithmetic or algorithmic means to calculate said looseness interval.

A further embodiment comprises storing the generated pixel-wise looseness intervals.

A further embodiment comprises outputting grayscale, high addressability, image statistics or templates match identifiers generated by said determining step.

A further embodiment comprises in said fitting step fitting said plurality of templates either in a parallel or serial process.

[0035] Loose fit generates a template match identifier that may be used in subsequent processing modules, such as a module that would determine an enhanced output signal.

[0036] Accordingly, in some embodiments the system and methods of embodiments of this invention provide a method to decrease the cost of printer electronic hardware to practical levels for example by designing a filter possessing a relatively small number of gray-scale templates.

[0037] These and other features and advantages of the system and methods of this invention are described in or are apparent from the following detailed description of the drawings and of the preferred embodiments.

BRIEF DESCRIPTION OF THE DRAWINGS

[0038] Fig. 1 is a diagram illustrating a conventional template matching filter paradigm.

[0039] Fig. 2A shows a flowchart of a conventional binary template matching arrangement.

[0040] Fig. 2B shows a diagram of an example of a series of templates used in a binary matching template arrangement.

[0041] Fig. 3A shows a representation of a 3x3 window and the positions of pixels ($x_1,...,x_9$).

[0042] Fig. 3B shows a diagram of a gate structure of the template matching operation for the templates of the example in Fig. 3A.

[0043] Figs. 4A-C show a series of diagrams of an example of an ideal image, an observed image with holes and breaks and an image resulting from a template matching operation on the observed image.

[0044] Fig. 5A shows a diagram of a system configuration using a template matching filter system according to the prior art.

[0045] Fig. 5B shows another diagram of a system configuration using a template matching filter system of the present invention.

[0046] Fig. 6 is a diagram illustrating a gray-scale matching configuration in accordance with the present invention.

[0047] Fig. 7 is a flowchart illustrating a serial loose-gray-scale matching system in accordance with the present invention.

[0048] Fig. 8 is a detailed flowchart illustrating a serial test for loose-gray-scale template matching of a family of templates in accordance with the present invention.

[0049] Fig. 9 is a flowchart illustrating a parallel test for a loose-gray-scale template match with one template in accordance with the present invention.

[0050] Fig. 10 is a diagram illustrating a parallel test for a family of templates performing matching operations using loose-gray-scale template matching in accordance with the present invention.

[0051] Fig. 11 is a diagram of multiple families of templates performing a template matching operation using loose-gray-scale template matching in accordance with the present invention.

DESCRIPTION OF THE PREFERRED EMBODIMENT

[0052] For a general understanding of the present invention, reference is made to the drawings. In the drawings, like reference numerals have been used throughout to designate identical elements. In describing the present invention, the following term(s) have been used in the description.

[0053] The term "bitmap" refers herein to a digital image quantized in a variety of forms, such as binary, 8 bits/pixel, or some intermediate number of bits/pixel, or some high resolution quantization state.

[0054] The term "data" refers herein to physical signals that indicate or include information. When an item of data can indicate one of a number of possible alternatives, the item of data has one of a number of "values." For example, a binary item of data, also referred to as a "bit," has one of two values, interchangeably referred to as "1" and "0" or "ON" and "OFF" or "high" and "low." A bit is an "inverse" of another bit if the two bits have different values. An *N*-bit

item of data has one of 2^N possible values.

[0055] The term "data" includes data existing in any physical form, and includes data that are transitory or are being stored or transmitted. For example, data could exist as electrical, electromagnetic or other transmitted signals or as signals stored in electronic, magnetic, or other form. The terms image signal, video data, and pixel are interchangeably used herein to describe discrete digital signals that represent the active (on) or inactive (off) state of an element within a digital image. In addition, shaded or crosshatched portions of image bitmaps depicted herein are intended to represent black or active pixels (having a value of 1 in a binary representation) within the bitmaps. Such a representation is not intended to limit the present invention, but to simplify the description thereof. Accordingly, the present invention may be operated in the same manner by substituting white pixel states wherever black pixels are indicated, and vice versa.

[0056] "Circuitry" or a "circuit" is any physical arrangement of matter that can respond to a first signal at one location or time by providing a second signal at another location or time. Circuitry "stores" a first signal when it receives the first signal at one time and, in response, provides substantially the same signal at another time. Circuitry "transfers" a first signal when it receives the first signal at a first location and, in response, provides substantially the same signal at a second location. An example of a circuit is a data or address bus in an electromechanical system such as a printing system or computer.

[0057] A "data storage medium" or "storage medium" is a physical medium that can store data. Examples of data storage media include magnetic media such as diskettes, floppy disks, and tape; optical media such as laser disks and CD-ROMs; and semiconductor media such as semiconductor ROMs and RAMs. As used herein, "storage medium" covers one or more distinct units of a medium that together store a body of data.

[0058] "Memory circuitry" or "memory" is any circuitry that can store data, and may include local and remote memory and input/output devices. Examples include semiconductor ROMs, RAMs, and storage medium access devices with data storage media that they can access. A "memory cell" is memory circuitry that can store a single unit of data, such as a bit or other n-ary digit or an analog value.

[0059] A "data processing system" is a physical system that processes data. An "image processing system" is a data processing system that processes image data. A "data processor" or "processor" is any component or system that can process data, and may include one or more central processing units or other processing components.

[0060] An "array of data" or "data array" or "array" is a combination of items of data that can be mapped into an array. A "two-dimensional array" is a data array whose items of data can be mapped into an array having two dimensions.

[0061] An item of data "defines" an array when it includes information sufficient to obtain or produce the array. For example, an item of data defining an array may include the defined array itself, a compressed or encoded form of the defined array, a pointer to the defined array, a pointer to a part of another array from which the defined array can be obtained, or pointers to a set of smaller arrays from which the defined array can be obtained.

[0062] "Control circuitry" is circuitry that provides data or other signals that determine how other components operate. For example, "instruction circuitry" is control circuitry that provides items of data indicating instructions to a component that includes processing circuitry. Similarly, "addressing circuitry" is control circuitry that provides items of data indicating addresses to a component that includes memory circuitry.

[0063] Control circuitry provides signals that "control" transfer of data by bus circuitry if the signals determine sources and destinations of the transfers of data by the bus circuitry. For example, the control circuitry could provide signals to a source so that it provides an item of data to the bus circuitry; the control circuitry could also provide signals to one or more destinations so that they receive the item of data from the bus circuitry.

[0064] An "image" may include characters, words, and text as well as other features such as graphics. A text may be included in a set of one or more images, such as in images of the pages of a document. An image may be divided into "segments," each of which is itself an image. A segment of an image may be of any size up to and including the whole image.

[0065] An item of data "defines" an image when the item of data includes sufficient information to produce the image. For example, a two-dimensional array can define all or any part of an image, with each item of data in the array providing a value indicating the color of a respective location of the image.

[0066] Each location in an image may be called a "pixel." Hence, a "pixel" is the smallest segment into which an image is divided or accessed in a given system. In an array defining an image in which each item of data provides a value, each value indicating the color of a location may be called a "pixel value". Each pixel value is, for example, a bit in a "binary form" of an image, a gray scale value in a "gray scale form" of an image, or a set of color space coordinates in a "color coordinate form" of an image, the binary form, gray scale form, and color coordinate form each being a two-dimensional array defining an image. The invention will hereafter be described in terms of a single-color gray-scale embodiment, however, there is no intent to limit the invention to such a single-color system as it has application to multicolor systems as well.

[0067] An item of data "relates to" part of an image, such as a pixel or a larger segment of the image, when the item of data has a relationship of any kind to the part of the image. For example, the item of data could define the part of the image, as a pixel value defines a pixel; the item of data could be obtained from data defining the part of the image;

the item of data could indicate a location of the part of the image; or the item of data could be part of a data array such that, when the data array is mapped onto the image, the item of data maps onto the part of the image.

[0068] An operation performs "image processing" when it operates on an item of data that relates to part of an image.

[0069] Pixels are "neighbors" or "neighboring" within an image when there are no other pixels between them or they meet an appropriate criterion for neighboring. If, for example, the pixels using adjoining criteria are rectangular and appear in rows and columns, each pixel may have 4 or 8 adjoining neighboring pixels, depending on the criterion used.

[0070] The "looseness interval" of the present invention is a measure of the difference between a template and an observed neighborhood of pixels. Each pixel can possess a looseness interval and a template and neighborhood can possess an overall looseness interval. In the pixel-wise case, the looseness interval could be a simple metric, such as the absolute value of the difference between an observed pixel value and a template value. More complicated intervals may also be defined, such as intervals that are dependent upon the sign of the difference, or are a function of the difference. The function may be arithmetic, such as multiplication by a factor, algebraic, such as being raised to a power, or some other algorithmic means to represent a degree of difference between the observed neighborhood of pixels and the template. An overall template looseness interval could be a combination of the pixel looseness intervals, or it could be a function directly applied to the observed pixel neighborhood and template, where that function provides a degree of difference. Example of the overall template looseness interval are averaging of at least a portion of the pixel looseness interval, and taking the maximum of the pixel looseness intervals. In this patent specification we often use looseness interval to denote either or both a pixel looseness interval and an overall template looseness interval.

[0071] The "threshold looseness interval" is the maximum allowable value for the looseness interval that indicates a loose-fitting match. As with the definition of looseness interval, threshold looseness interval may be defined pixel-wise or overall-template-wise.

[0072] Fig 5b shows grayscale image data 181 inputted to a digital printer 184. The printerincludes a line width enhancement circuit 182 and line width enhanced bitmap 183 output to a marking engine 185 to produce an enhanced printed 186. Line width enhancement in such a digital printer is typically performed for one or more of several possible reasons. One reason for the enhancement is compensation for marking process characteristics. For instance, a given marking process may not be able to print single pixel lines, and therefore it is desired to grow those lines in the digital image prior to printing to enable their representation on the final printed image. Another reason for line width enhancement is user preference, where a given user may prefer a particular darkness or thickness of character strokes, and an enhancement operation transforms the strokes from a received width to the preferred width.

[0073] Fig. 6 shows one embodiment of a generalized functional block diagram of a loose-gray-scale template matching filter generation system 250 according to this invention. The template matching system 220 is connected to a template set and receives as input the gray-scale image 200 over a data buffer 210. The gray-scale image can also be received over a line signal or other link. The gray-scale image buffer provides a variety of image data to the loose-gray-scale template matching system 220.

[0074] In general the data source can be any one of a number of different data sources as an example such a source can be a scanner, digital copier, digital camera or any known device suitable for electronic source generation and/or storing or transmitting the electronic image. Further, the data source can be multiple devices hooked up serially or in parallel with the template matching system.

[0075] The data received from the data source is input to the loose-gray-scale template matching module, where one or more sets of templates are applied to the image. The signal output from the template matching block could be a match identifier, which could be passed onto a module for final signal selection, or the matching module itself could generate the final output signal by outputting the output signal associated with a matched template. The output signal could be a variety of forms such as a gray-scale pixel value, a binary pixel value, a group of binary of gray-scale pixel values, or an item of data describing the match condition.

[0076] With reference to Fig. 7, a process for a serial implementation for matching multiple templates is shown. The loose-gray-scale template matching system includes an input image which is the output of a data source and a set of templates 300. The input image 310 is subsequently connected to locate target pixel circuit 320 for determining a target pixel. Surrounding the target pixel 330, a window is designated so as to extract a defined portion of the image about the target pixel 330. Such a window could be a 3x3, 4x4, 7x4...., etc. type of matrix. Loose-gray-scale templates are stored in a template storage module 300. A template from the template storage module is input to the template comparison module 340. A loose-gray-scale template match is performed 340 on the defined portion of the image data. If a loose match occurs a fit decision is created. A fit decision could be a code that identifies the matching template, or it could be some resultant value, such as an output value or statistic of the image. If a loose match does not occur with the given template then the next template from the set of templates 300 is tested for a loose match, until there are no templates 350 to loosely match in the set 300. If none of the templates matched a different fit decision is defined, such as "do not change the value of the input image 310". An example of a fit decision is an address that could be used to access an output value stored in a look-up table. The address could be a digital value formed of several binary bits. If a fit does not occur the NO decision is passed back to the template storage module where another template may be

accessed for a subsequent matching test. When all the templates are applied the image processing operation could return to the original image and extract a different neighborhood of observed pixels for further matching tests. Note that the matching tests could proceed until all templates are examined to create a collection of fit decisions, or the matching operation could have concluded upon obtaining a successful match. A process for a serial implementation for matching multiple templates is shown.

[0077] Mathematically equivalent matching processes could be employed in indirect matching operations, these would for example include match identifiers which represent a window of image data of the received image and match identifiers representing the templates and associated threshold looseness intervals.

[0078] Fig. 8 further teaches methods with a serial implementation of loose grayscale template matching. With reference to Fig. 8, in another embodiment of the present invention a serial loose-gray-scale template matching system is shown where first a target pixel is initialized 400. About the target pixel, a set of template values is read in 410 and a particular region of support for a current template is determined. Another way of looking at this concept is a set of templates is held in memory, and successive windowed neighborhoods are compared to the stored templates. From the region of support an extraction of the group of coincidental pixel values from the image relative to the target pixel is performed 430. A test for a loose-gray-scale template fit is performed 440 and if the result is NO then the sequence proceeds to the next template 450 until all the templates are tested 460. If the test of the loose-gray-scale match is YES then an action based on loose-gray-scale match is generated for the target pixel 470.

[0079] In reference to Fig. 9, there is shown an embodiment of a parallel test 800 functionally denoted by f for a single template T performing a matching operation using loose-gray-scale template matching according to the present invention. In particular, the present invention receives a series of input pixels $(x_1, x_2, ..., x_N)$ 810 where N is the size of the neighborhood in number of pixels and applies a loose-gray-scale match

[0080] Starting with the definition of loose-gray-scale template matching, several implementation and processing schemes using sets or multiple sets of templates will be described below.

[0081] For a single template T there is a loose match between the digital image in the observation window and the template when each pixel of the observation window, has an absolute value difference 820 between the image value at each pixel x_i and the template value for each pixel t_i that is less or equal 830 to the chosen looseness interval threshold δ_i. The aforesaid absolute value difference in this embodiment is the looseness interval for each pixel in the window, and the limiting value for looseness acceptability for a match at each pixel is the looseness interval threshold. A loose-gray-scale template is composed of the template pixels values t_i and the threshold intervals δ_i, or a single threshold interval for the template In this embodiment we employ looseness intervals and looseness interval thresholds for each pixel in the window, and all, or some predefined number of looseness intervals must be less than the threshold looseness interval to define a loose match. The individual looseness interval decisions are evaluated in concert in combine block 840 to generate an overall fit decision 850.

[0082] The template fitting process may be formulated as shown in Eqs. 2 and 3 where it is chosen to define a loose fit as having all pixel looseness intervals possess values less than or equal to their respective looseness interval thresholds,

$$(2) \qquad f(x_1, x_2,...,x_N;T) = f(x_1,x_2,...,x_N; t_1, t_2,...,t_N)$$

$$=\min \{I_{(0,.\delta i]} |x_i - t_i| ; i = 1,...,N\}$$

where

$$(3) \qquad I_{(0.\delta i]}(u)= 1 \text{ if } 0 \le u \le \delta_i \text{ and } 0 \text{ otherwise}$$

and where I is called the characteristic function or indicator function and | | is the absolute value.

[0083] We have expressed a loose-fitting template as a set of pixel values and corresponding threshold looseness intervals. There are alternative representations, such as two templates, one that specifies the lower bound that pixels may possess and the other template specifies the upper bound that pixels may possess to qualify as a loose-fitting match. That representation can be written as the following equation,

$$(4) \qquad f(x_1,x_2,...,x_N;T) = f(x_1,x_2,...,x_N; t_1, t_2,...,t_N)$$

$$=\min [I_{[ai,bi]} (x); i = 1,...,N]$$

where

$$(5) \qquad I_{[a_i,b_i]}(u) = 1 \text{ if } a_i <= u <= b_i \text{ and 0 otherwise}$$

One advantage of this representation is that the threshold looseness interval is not limited to be symmetric about a central value.

[0084] There are other mathematical means to express the matching condition as described in Eqs. 2 and 3. One such equivalent form that we have found useful is provided in Eq. 6,

$$(6) \qquad f(x_1,x_2,...,x_N;T) = f(x_1,x_2,...,x_N;\, t_1,\, t_2,...,t_N)$$

$$= \min\left[I_{[d_i,s_i]}(x_i);\ i = 1,...,N\right]$$

Where

$$(7) \qquad s_i = t_i + \delta_i \qquad d_i = t_i - \delta_i$$

Hence, there are a multitude of means to compute loose matches according to the loose-fitting criteria of the present invention. Another example is that loose matching could be determined by performing exact matching operations at lowered quantization, that is, an exact match using only the upper three bits of an eight bit pixel value is a loose match with respect to the pixel value at 8 bits. It is the intention of this patent to cover all such equivalent forms of the loose-fitting criteria.

[0085] Note that alternative looseness criteria may be applied. For instance, an overall template looseness interval may be computed by processing the individual pixel looseness intervals with an operation such as averaging a selected group, or taking their collective maximum value. That overall template looseness interval could then be compared to an overall threshold looseness interval to establish a match or no match condition.

[0086] In Fig. 10, there is illustrated another embodiment of the present invention of a parallel test for loose-gray-scale template matching for a set of templates functionally denoted as h. Template matching filters will usually be composed of multitude of template, which refer to as the template set or a template family. Here, the values for each template are stored into memory as well as the corresponding threshold looseness intervals. In the figure the template values are channeled simultaneously to the processing step where a looseness fit is ascertained for each template in parallel. A fit decision is generated for each loose match test and then the information is combined in a manner to reach an overall fit decision of whether a particular template loosely matched or not. By performing a parallel loose match operation more decision making steps can be performed in a shorter interval of time. Parallel implementation is a preferred embodiment in applications implemented in hardware, such as an Application Specific Integrated Circuit (ASIC), whereas serial implementations as shown in Fig. 7 are typically preferred for software implementations.

[0087] In a template-wise parallel implementation, for a collection of M templates, we can say that there M loose matching template operators $f(*,T_i)$ $i = 1,...,M$, operating in parallel for a set of templates (T_1, T_2, ... T_M). The results of each individual template match decision are combined in a manner so as to generate the overall template set fit decision. Fig. 10 illustrates an embodiment of this template-parallel method where a serial input of digital image data is converted in parallel and combined to make the overall template fit decision. The combine block 740 can take a variety of forms. If the template fit decisions are disjoint, that is, only one template can loosely fit, the combine block can simply select the match decision that yielded a match. If the match decisions are not disjoint, the combine block must in some cases arbitrate between multiple matches. In this case we have found it beneficial to give priority to certain templates to allow the combine block to select a match of higher priority. Useful priority criteria have been based on template size (number of pixels) and threshold looseness intervals. A smaller threshold looseness interval would tend to indicate a more exact match and would be given priority in typical applications. Note that other related arbitration criteria fall within the scope of the present invention.

[0088] Also, various templates may be grouped in sets wherein each set serves a particular processing task. Within each group, the templates may be arranged or a general arbitration procedure may be defined. The arbitration procedure may be a logical expression or a functional combination of the individual loose matching results. A family-parallel embodiment of this concept is shown in Fig. 11. In Fig. 11 the pixel values of the observed neighborhood are inputted

to multiple parallel matching channels, where each will apply a filter defined by a family, or set, where matching with a set of templates is shown in Fig. 10.

[0089] It should be noted that templates sets and threshold looseness intervals could be modified during the processing of an image. For instance, tag information indicating image type could select a particular set of templates to be applied to that image type, or select a threshold looseness interval that is suitable for that image type. Further, prior processing history could be incorporated into loose-gray-scale template matching. For instance, although it has been described to utilize window pixel values solely from the observed image, recursive methods could be employed where the window of pixels applied to the matching operations could contain some mix of observed pixels values and recently processed pixel values. The above discussion simplifies the teaching of the general method by referring to either of these windowing methods as extracting observed pixels.

[0090] Further, pending U.S. patent application No. 09/222,919, filed December 30, 1998, describes a method of "vector template matching" where some observed values may be extracted from other dimensions. An example is the extraction of pixels values from multiple frames in a time sequence, multiple dimensions in a color space representation, and multiple dimensions in space. In the present invention, observed pixels may be extracted from such multi-dimensional sources.

Claims

1. A template fitting method for fitting a plurality of templates in correspondence with a gray-scale image,, comprising:

   receiving gray-scale image information of an observed image;
   defining a window of said gray-scale image information so as to enable a loose-gray-scale fit between said defined window and said plurality of templates;
   extracting a plurality of pixels of said image information using said window to which to fit at least one of the plurality of templates; and
   fitting one of the plurality of templates to said defined window within a threshold looseness interval associated with the template wherein a looseness interval between the window of image data and the template is less than the threshold looseness interval.

2. The template fitting method of **claim 1**, wherein the fitting step comprises:
   producing an item of data which represents a fit or no fit condition.

3. A multiple template fitting method including a plurality of single templates or plurality of families of templates for correspondence with a gray-scale image , comprising:

   receiving image information of an observed image;
   defining a window of said image information so as to enable a loose grayscale fit between said defined window and said plurality of singular or families of templates;
   fitting one of the plurality of singular or families of templates to said defined window within a threshold looseness interval; and
   outputting an item of data indicating a result of a correspondence between a single or family of templates and said received image information within a threshold looseness interval.

4. The template fitting method of **claim 3**, comprising:
   wherein said fitting step fitting said plurality of templates either in a parallel or serial process.

5. The template fitting method of **claim 3**, comprising:
   wherein said fitting step using an arithmetic or algorithmic process to calculate said looseness interval.

6. A method for matching a plurality of templates with a received image wherein said received image being bitmap data including at least a plurality of gray-scale pixel tiles that define said image, said matching method comprising:

   receiving said image data so as to extract pixel tile information of said received image wherein said pixel tile information being of a predetermined matrix size;
   matching loosely said pixel tile information with at least one of a plurality of templates so as to generate pixel-wise looseness interval values therebetween; and
   outputting a portion of enhanced said pixel tile information wherein said enhanced pixel tile information being

formed by a matching of a template with pixel-wise looseness values.

7. The method of **claim 6** further comprising:
said matching step using either a template or a family of templates.

8. The method of **claim 6** further comprising:
said matching step using template-wise looseness intervals instead of said pixel-wise looseness interval values .

9. A loose-gray-scale template matching method including at least an initial point for locating within the received image a plurality of pixel points surrounding said initial point, said method comprising the steps of:

locating said initial point and said plurality of pixel points,
generating a looseness interval about said located initial point with template information associated therewith,
determining which one of a plurality of templates for fitting fits within a threshold looseness interval, and
outputting an item of data indicative of whether a template is within the looseness threshold thereby indicating that the template is in agreement therewith.

10. The method of **claim 9** comprising;
said determining step determining a looseness interval of a pixel-wise looseness.

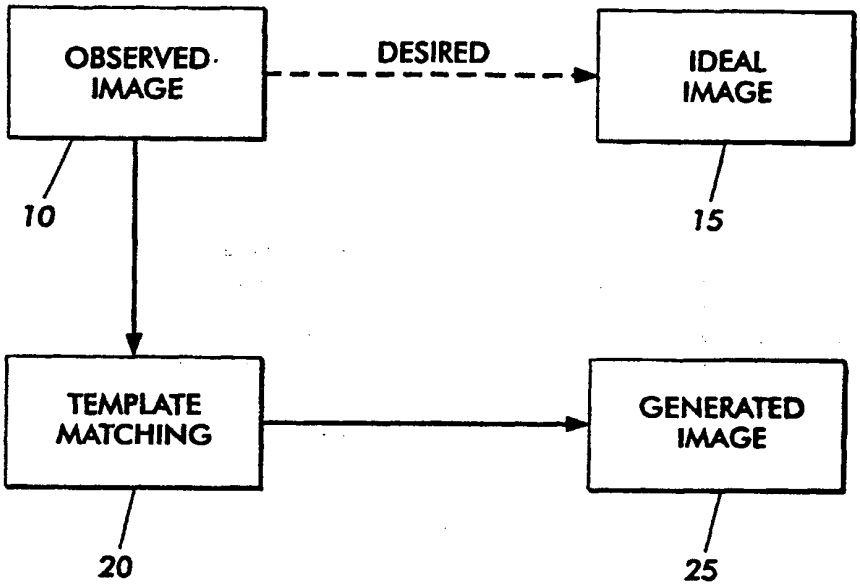
OBSERVED IMAGE
DESIRED
IDEAL IMAGE
10
15
TEMPLATE MATCHING
GENERATED IMAGE
20
25

FIG. 1

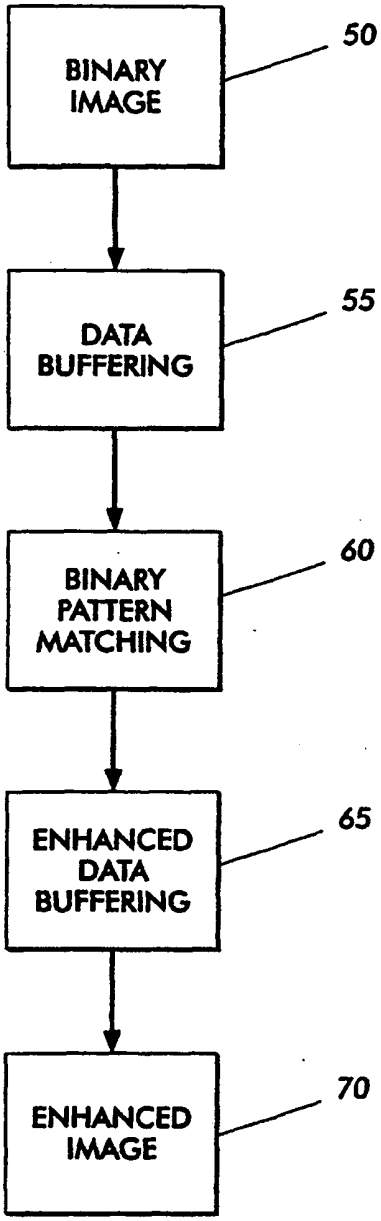

FIG. 2A
PRIOR ART

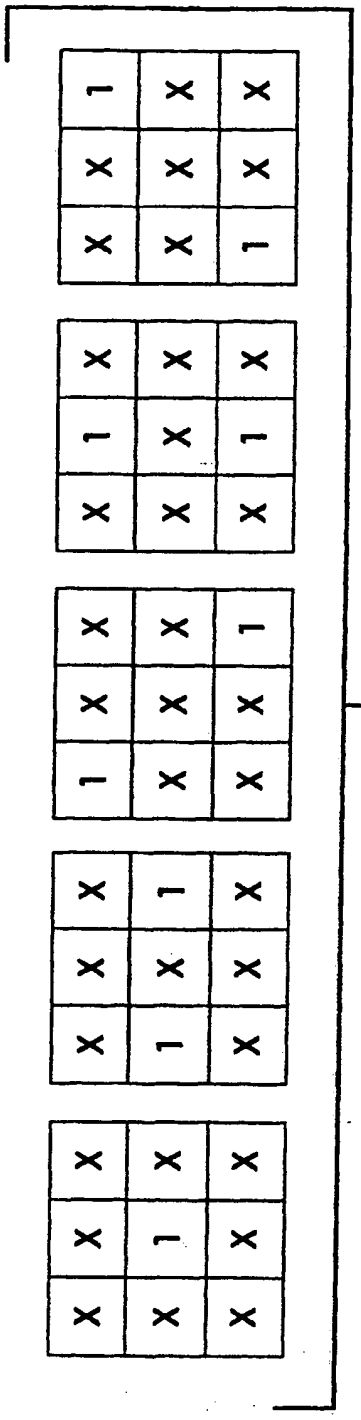

FIG. 2B
PRIOR ART

X_1	X_2	X_3
X_4	X_5	X_6
X_7	X_8	X_9

FIG. 3A

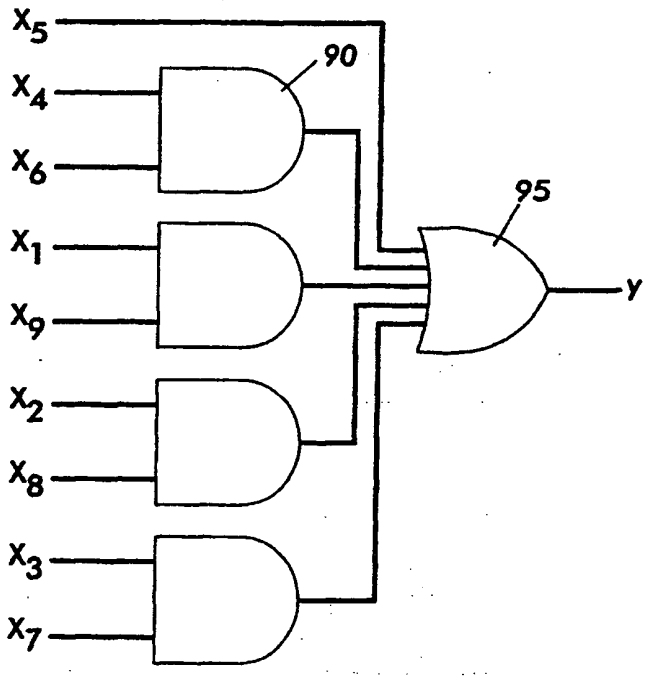

FIG. 3B

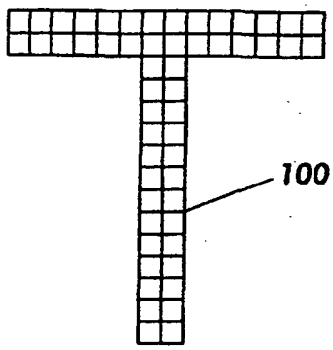

FIG. 4A

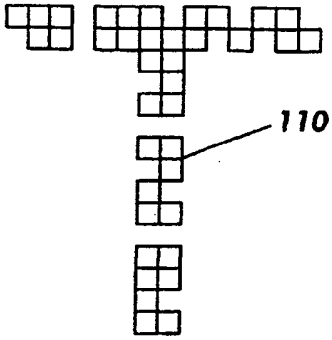

FIG. 4B

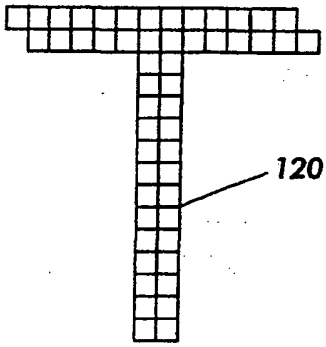

FIG. 4C

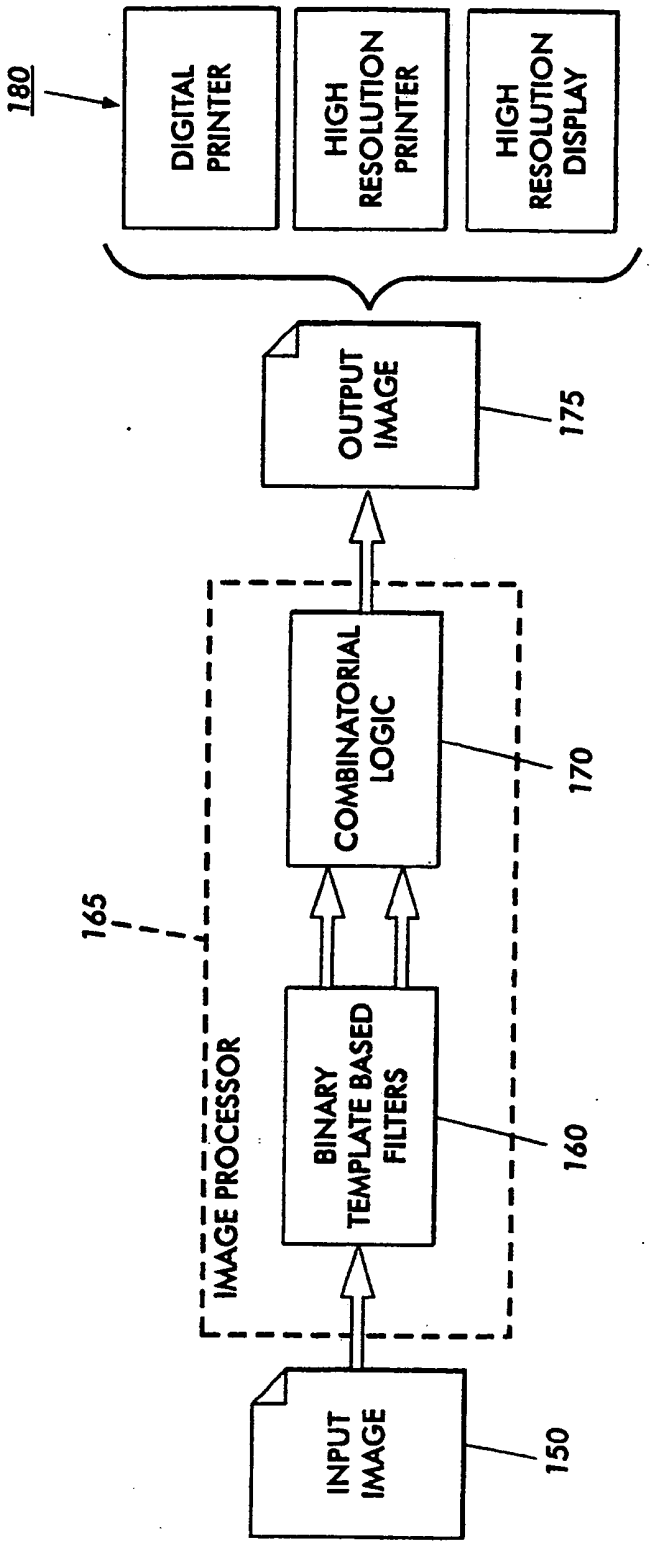
180
DIGITAL PRINTER
HIGH RESOLUTION PRINTER
HIGH RESOLUTION DISPLAY
OUTPUT IMAGE
175
COMBINATORIAL LOGIC
170
165
IMAGE PROCESSOR
BINARY TEMPLATE BASED FILTERS
160
INPUT IMAGE
150

FIG. 5A

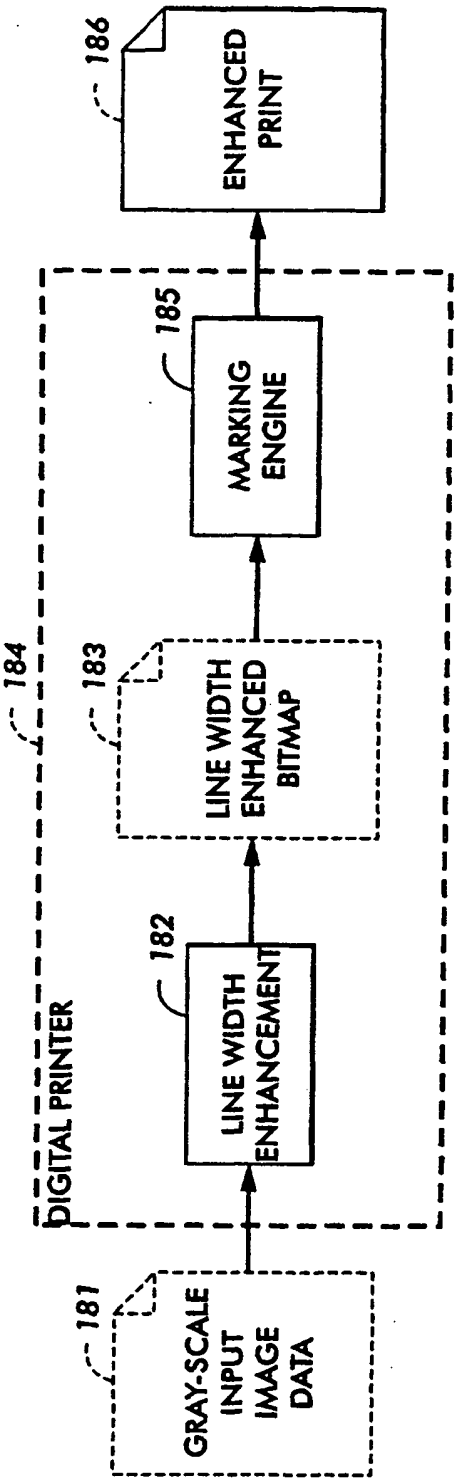
186
ENHANCED PRINT
185
MARKING ENGINE
184
183
LINE WIDTH ENHANCED BITMAP
182
LINE WIDTH ENHANCEMENT
DIGITAL PRINTER
181
GRAY-SCALE INPUT IMAGE DATA

FIG. 5B

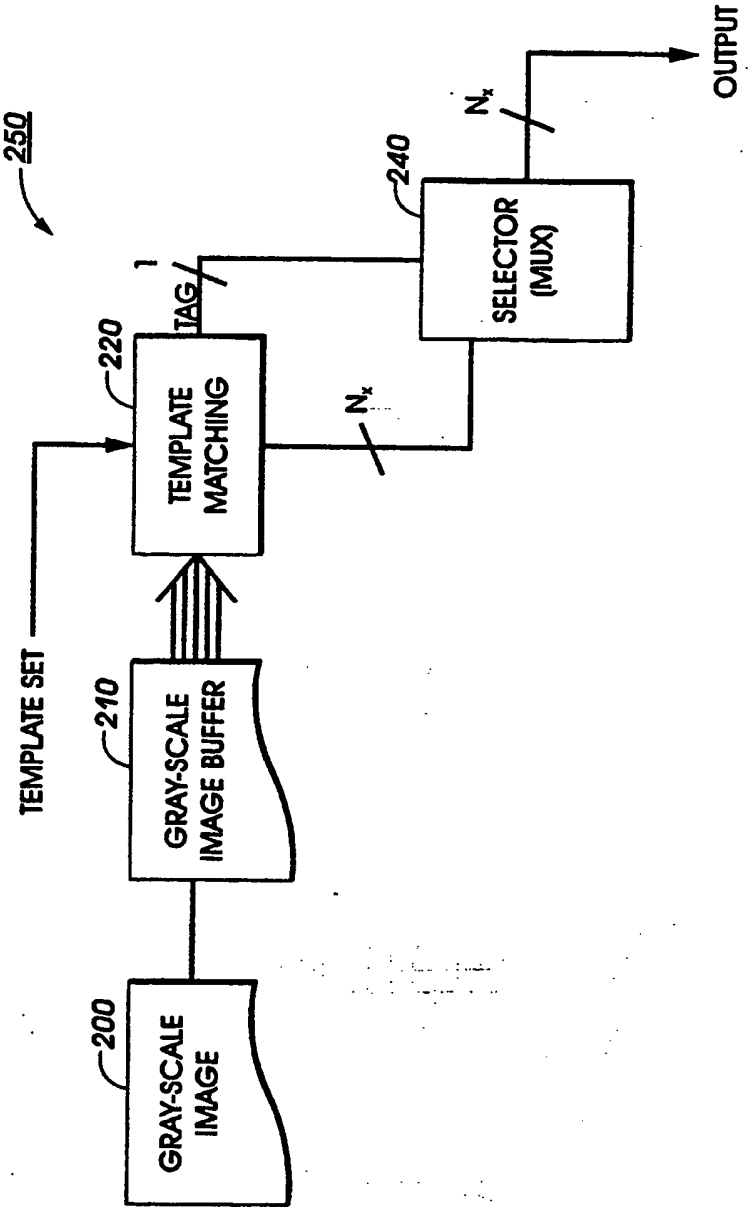
250
200
GRAY-SCALE IMAGE
210
GRAY-SCALE IMAGE BUFFER
TEMPLATE SET
220
TEMPLATE MATCHING
TAG
1
N_x
240
SELECTOR (MUX)
N_x
OUTPUT

FIG. 6

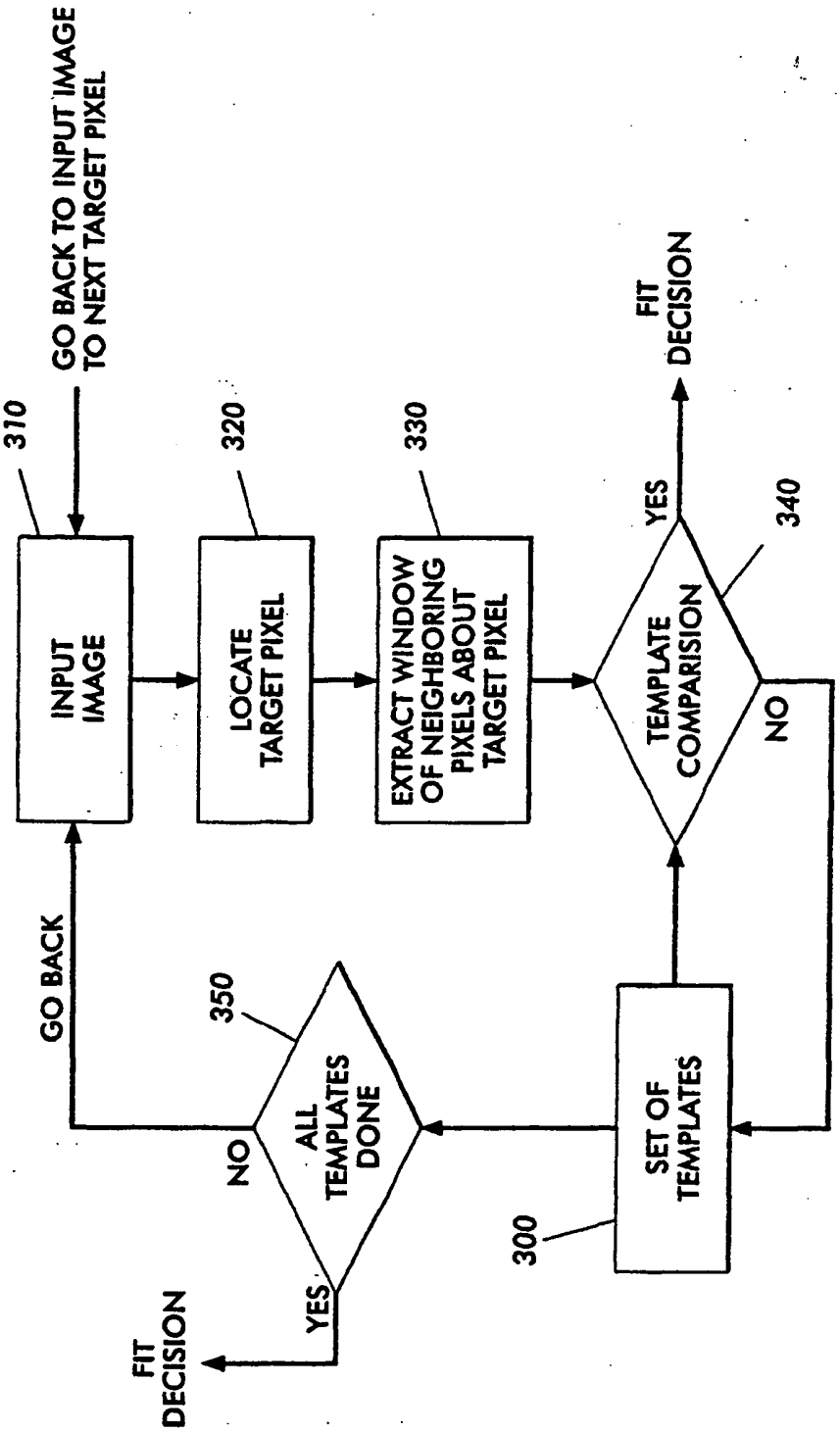
GO BACK TO INPUT IMAGE TO NEXT TARGET PIXEL
310
INPUT IMAGE
320
LOCATE TARGET PIXEL
330
EXTRACT WINDOW OF NEIGHBORING PIXELS ABOUT TARGET PIXEL
340
TEMPLATE COMPARISION
YES
FIT DECISION
NO
SET OF TEMPLATES
300
350
ALL TEMPLATES DONE
NO
GO BACK
YES
FIT DECISION

FIG. 7

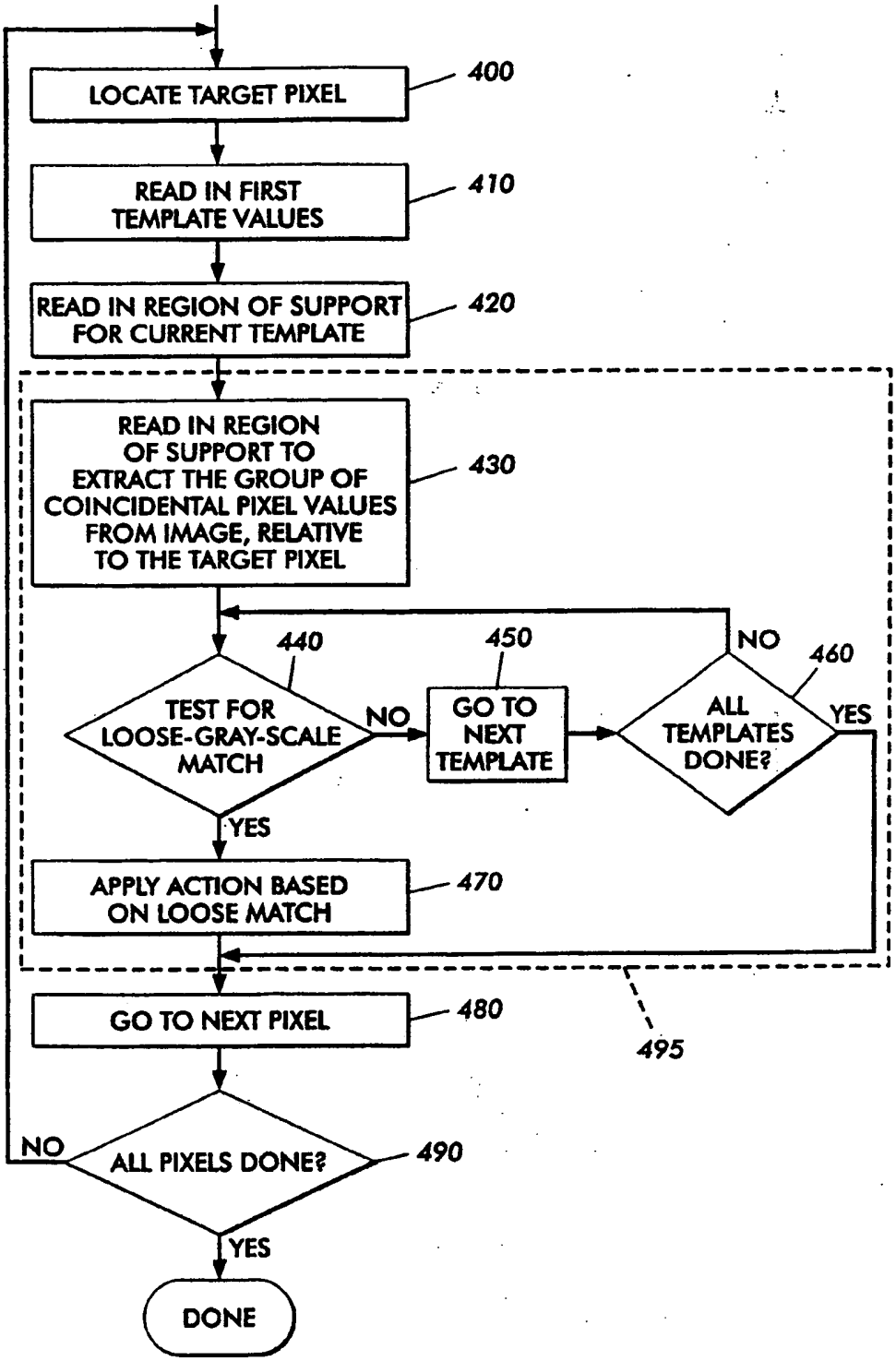
LOCATE TARGET PIXEL
400
READ IN FIRST TEMPLATE VALUES
410
READ IN REGION OF SUPPORT FOR CURRENT TEMPLATE
420
READ IN REGION OF SUPPORT TO EXTRACT THE GROUP OF COINCIDENTAL PIXEL VALUES FROM IMAGE, RELATIVE TO THE TARGET PIXEL
430
440
TEST FOR LOOSE-GRAY-SCALE MATCH
NO
450
GO TO NEXT TEMPLATE
460
ALL TEMPLATES DONE?
NO
YES
YES
APPLY ACTION BASED ON LOOSE MATCH
470
495
GO TO NEXT PIXEL
480
NO
ALL PIXELS DONE?
490
YES
DONE

FIG. 8

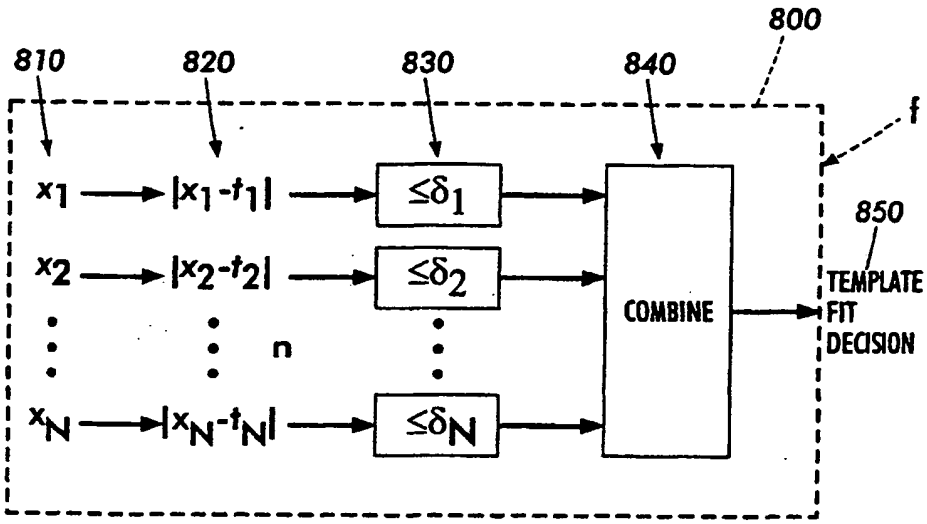
800
810
820
830
840
f
x_1
$|x_1-t_1|$
$\leq\delta_1$
x_2
$|x_2-t_2|$
$\leq\delta_2$
n
x_N
$|x_N-t_N|$
$\leq\delta_N$
COMBINE
850
TEMPLATE FIT DECISION

FIG. 9

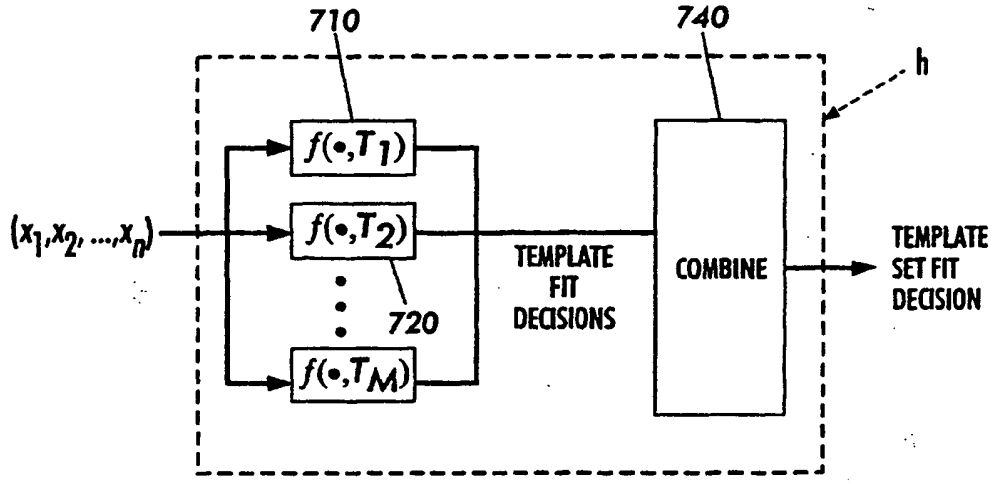
710
740
h
$f(\bullet,T_1)$
$f(\bullet,T_2)$
$f(\bullet,T_M)$
720
$(x_1,x_2,\ldots,x_n)$
TEMPLATE FIT DECISIONS
COMBINE
TEMPLATE SET FIT DECISION

FIG. 10

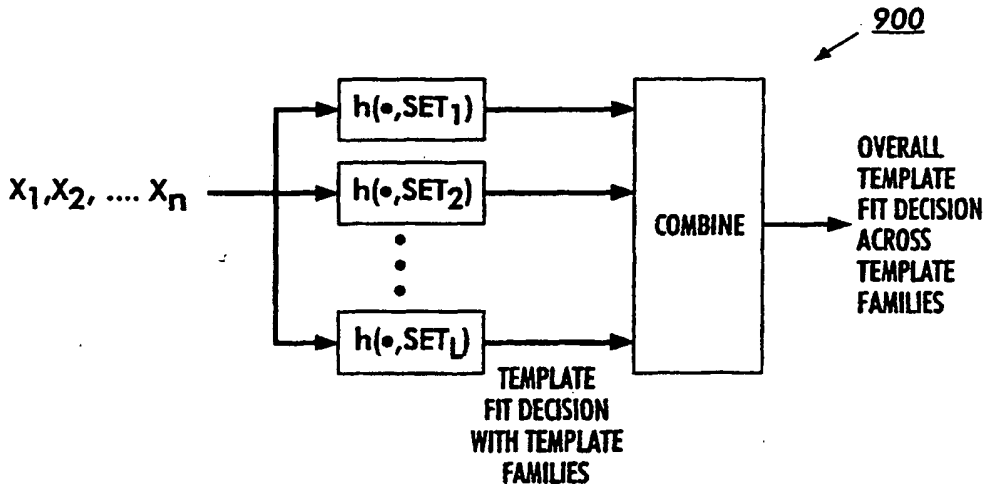
900
$h(\bullet,SET_1)$
$h(\bullet,SET_2)$
$h(\bullet,SET_L)$
$x_1,x_2,\ldots x_n$
COMBINE
OVERALL TEMPLATE FIT DECISION ACROSS TEMPLATE FAMILIES
TEMPLATE FIT DECISION WITH TEMPLATE FAMILIES

FIG. 11

This Page is Inserted by IFW Indexing and Scanning Operations and is not part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☐ FADED TEXT OR DRAWING
- ☐ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☑ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☑ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.
As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.